

Estimaciones

miércoles, 4 de septiembre de 2024 01:17

ESTIMACIONES DE SOFTWARE EN GENERAL

Después se van a plantear estimaciones Agiles

Lo más difícil de un producto de software es saber que queremos construir.

Lo segundo más difícil es hacer una estimación

Estimación:

- Definir el tiempo que nos puede llegar hacer una actividad. Estimamos teniendo en cuenta los recursos que tenemos y cuanto nos va a costar, también se tiene en cuenta a las capacidades de cada recurso en base a la complejidad de lo que queremos realizar.
 - Lo primero que se llega estimar, es el **ALCANCE**, ósea el **TAMAÑO** del producto que el cliente espera.
 - Lo primero que se utilizó para esbozar estimación para el tamaño de software fue las líneas de código sin comentar. Esto no funciona

La primera parte del Proceso de Estimación es:

Estimar Tamaño: Esto lo vamos hacer eligiendo con que unidad de medida lo vamos hacer. El **"QUE"**

Estimar Esfuerzo: Esto lo medimos con horas Ideales(Se despeja el tiempo perdido) y lineales (Asumiendo que no vamos hacer Multitarea, que no va a ver trabajo en paralelo). El **"COMO"**

Estimar Calendario: El tiempo es la estimación del calendario. El **"CUANDO"**

Estimación Costo: Cuando va a salir. El **"Cuanto"**

Algunas consideraciones

Por definición una estimación no es precisa.

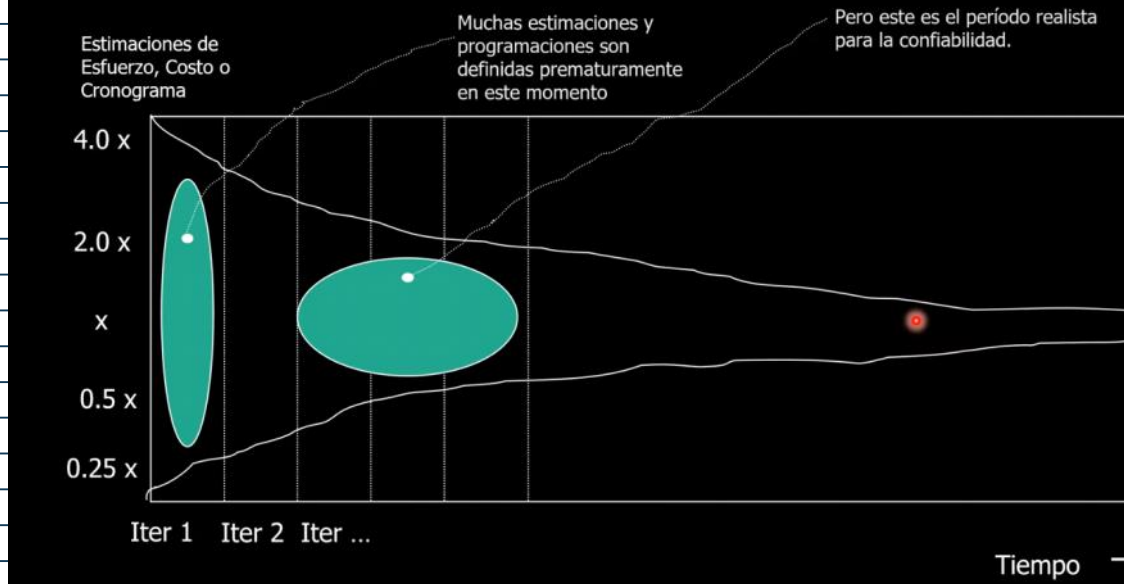
Estimar no es planear y planear no es estimar.

Las estimaciones son la base de los planes, pero los planes no tienen que ser lo mismo que lo estimado.

A mayor diferencia entre lo estimado y lo planeado mayor riesgo.

Las estimaciones no son compromisos.

¿Para qué estimamos?



¿De dónde vienen los Errores de Estimación?

Siempre le vamos a errar. Esto teniendo más frecuencia en:

Es mejor siempre estimar, que no hacerlo. Por más que le erremos. Esto porque se trata de compensar el error de las estimaciones con otras cosas

- Las actividades Omitidas:
 - Estimamos solamente el tiempo de Programación.
- Reuniones con los Clientes
- Los tiempos que hacen falta para la Gestión de Proyectos
- El procesos mismo de Estimación, Todos tienen un sesgo. Un margen de error
- Falta de Información
- No tenemos claro el proceso que vamos a utilizar para trabajar.

Entonces tenemos el Desafío de encontrar una manera de medir. Sobre todo el **Tamaño**.

Métodos que se utilizan:

- **Basados en la Experiencia**

- Datos Históricos

Significa que parto de la idea, de la experiencia de otros equipos, en otros proyectos a mí me puede servir. Entonces se recopila información de los proyectos que vamos cerrando y lo usamos como base para estimar los nuevos proyectos.

Puede elegir:

- El tiempo
- La tecnología
- El perfil de gente que trabajo
- Puede elegir los esfuerzos.

- Juicio Experto

No Existe un experto en todo

Pero se basa en los conocimientos del experto, por ende si desaparece el mismo la empresa deja de estimar.

Para obtener una duración, se utiliza una formula: $\text{Tiempo Optimista} + \text{Tiempo Pesimista} + 4 \text{ Tiempos Medios} / 6$

- Puro:

Un experto estudia las especificaciones y hace su estimación

se basa fundamentalmente en los conocimientos del experto
Si desaparece el experto, la empresa deja de estimar.

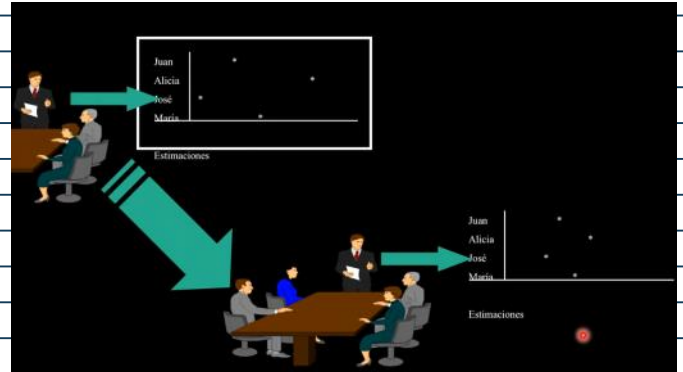
▪ **Delphi:**

Es la base del Poker Estimation.

En vez de tener un experto, tengamos un Grupo de expertos.

Se hacen las Estimaciones Individuales y después se presentan. Después se escuchan las estimaciones de todos y sobre esta base se corrige.

Es un método de iterativo Grupal, e individual



○ **Analogía**

- **Basados exclusivamente en los recursos**
- **Método basado exclusivamente en el mercado**
 - De acuerdo a la cara del cliente
- **Basados en los componentes del producto o en el proceso de desarrollo**
- **Métodos algorítmicos**
 - Apuntan a Formulas que te preguntan distintas variables (Complejidad, riesgo) y con una formula magica te da el tiempo.

ESTIMACIONES AGILES

NoEstimate?

- Es un enfoque, que sea a desarrollado dentro del ambiente ágil, que apunta y adhiere de que las estimaciones son desperdicio

Diferencia conceptual entre las estimaciones tradicionales con las Agiles

- Básicamente es: Estima el que hace el trabajo, luego estimamos como equipo.

Tips respecto de las Estimaciones

⚠ Si las estimaciones se utilizan como compromisos son muy peligrosas y perjudiciales para cualquier organización.

📊 Lo más beneficioso en las estimaciones es el “proceso de hacerlas”.

✓ La estimación podría servir como una gran respuesta temprana sobre si el trabajo planificado es factible o no.

🔧 La estimación puede servir como una gran protección para el equipo.

Estimaciones Agiles

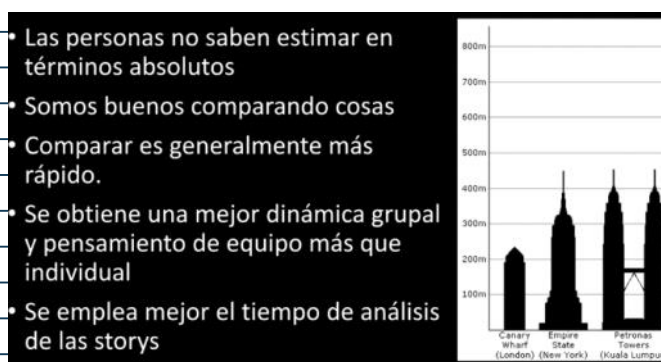
La estimación ágil es el proceso de estimar el tamaño o el esfuerzo de una tarea o proyecto utilizando metodologías ágiles.

Consiste en dividir un proyecto en partes más pequeñas y manejables y estimar el

tamaño o el esfuerzo necesario para cada parte. La estimación ágil suele realizarse en colaboración con el equipo, que incluye a quienes realizarán el trabajo.

A continuación, las estimaciones se utilizan para crear un plan de lanzamiento, que describe el orden en que se completarán y entregarán las piezas. La estimación ágil reconoce que el conocimiento es siempre incompleto y requiere que los planes se revisen a medida que el equipo aprende más sobre el producto que están construyendo. La estimación ágil es una parte importante de la planificación ágil, que se centra en aportar valor y establecer la confianza entre la empresa y los equipos del proyecto.

- Somos mejores haciendo estimaciones relativas (Story Points), no absolutas
 - Relativas : Quiere decir que somos mejores comparando
 - Esto para las Stories
 - Las Story Points no es una medida basada en tiempo, si no en los puntos de Historias



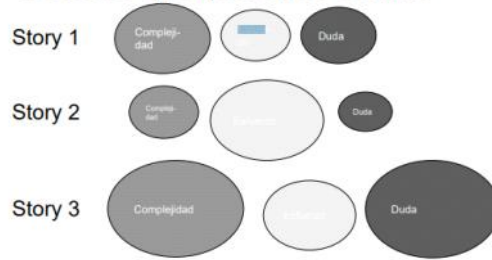
Poker Planner o Poker Estimation

- La unidad de Medida son los puntos de Historia
- Para hacer las comparaciones, utilizamos la **Canónica** (Que también se estima) como referencia para estimar las otras.
- Cuando se Estima, también se justifica porque lo estimamos así. Justificándolo a través de las 3 dimensiones que tiene la User Story
 - **Complejidad**
 - **Esfuerzo**: Cuantas horas ideales Lineales necesitamos para crear
 - Esfuerzo no es Tamaño
 - Esfuerzo no es Calendario
 - **Incertidumbre**: Que nivel de desinformación que tengo de esta Historia.
- Poker Estimation utiliza la Serie de Fibonacci.
 - Utiliza una escala Exponencial porque cuando se encuentran diferencias en las User, poder ver una notoriedad en esa diferencia identificada. Esto tiene un Crecimiento exponencial al igual que el Software.
 - Se calcula de la suma de las dos anteriores.

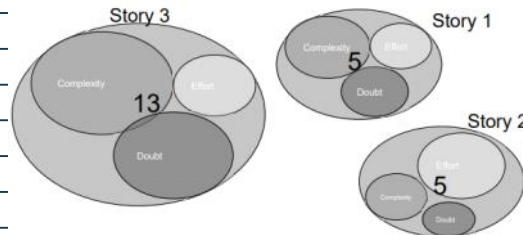
Que es una Story Point ?

- Es una unidad de medida específica (del equipo) de, complejidad, riesgo y esfuerzo, es lo que “el kilo” a la unidad de nuestro sistema de medición de peso
- Story point da idea del “peso” de cada story y decide cuan grande (compleja) es
- La complejidad de una feature/story tiende a incrementarse exponencialmente.

3 Stories que queremos estimar



Y si usamos números?



Velocidad

- La velocidad es una METRICA de Producto, mide Producto. Es la más Importante
 - Producto: Software Funcional
 - No se Estima, SE CALCULA.
 - Cuando ? Al final de la Sprind
- Velocidad/ Velocity es una medida (métrica) del progreso de un equipo.
 - Como se Calcula?:
 - Se calcula sumando todos los puntos de Historia, de las Users Story que el product Owner me acepto
 - Se cuentan los story poins de las Users Storys que están completas, no parcialmente completas.
 - Para que me Sirve ?
 - Sirve para ir viendo si logro "Desarrollo Sostenible (Principio de Manifiesto Agil)"
 - La Velocidad corrige los errores de estimación y nos ayuda a aprender en el camino.
 - Por mas que vengamos con una velocidad estable (Ejemplo 15 puntos en cada Sprind), no garantiza que en el sprin siguiente sea la misma velocidad
 - La velocidad también sirve para mostrar cuando producto funcional le estamos entregando al Product Owner
 - También para ver como va funcionando el equipo
 - Te sirve para estimar un poco el Sprind Siguiente.
- Tiene que venir con un contexto sobre esa velocidad. Ya que es la Realidad

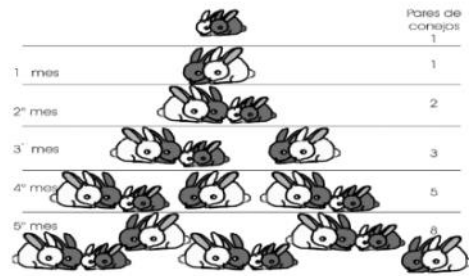
Una Propuesta de método de estimación

Poker estimation

- Popular entre los Agile practitioners, publicado por Mike Cohn •
- Combina opinión de experto, analogía y desegregación.
- Participantes en "planning poker" son desarrolladores
 - "Las personas más competentes en resolver una tarea deben ser quienes las estiman"

Fibonacci (se acuerdan?)

- La secuencia empieza en 1 y cada numero subsecuente es la suma de los dos precedentes. (1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144....)



Poker Planning

¿Cómo "decodificar" las estimaciones?

- Hay que Recordar el INVEST MODEL y que Una User hay que empezar y terminarla en una Iteracion y que el SMALL es una unidad relativa porque la User estan chiquita o tan grande (Deoendiendo de la capacidad del equipo y de la Duracion del Sprind que nosotros elegimos).

TODAS ESTAS COSAS, NOS DICE SI LA USER ES LO SUFICIENTEMENTE CHICA PARA PODER EMPEZARLA Y TERMINAR LA USER EN UNA ITERACION

- La recomendación es trabajar con User Chicas



- 0: Quizás ud. no tenga idea de su producto o funcionalidad en este punto.
- 1/2, 1: funcionalidad pequeña (usualmente cosmética).
- 2-3: funcionalidad pequeña a mediana. Es lo que queremos.
- 5: Funcionalidad media. Es lo que queremos
- 8: Funcionalidad grande, de todas formas lo podemos hacer, pero hay que preguntarse si no se puede partir o dividir en algo más pequeño. No es lo mejor, pero todavía
- 13: Alguien puede explicar por qué no lo podemos dividir?
- 20:Cuál es la razón de negocio que justifica semejante story y más fuerte aún, por qué no se puede dividir? ■ 40: no hay forma de hacer esto en un sprint.
- 100: confirmación de que está algo muy mal. Mejor ni arrancar.

Poker Planning

Prerrequisitos:

- Lista de features/stories a ser estimadas
- Cada estimador tiene un mazo de cartas.

Pasos:

1. Determine la **base story** (la canónica) que será usada para comprar con las otras stories. Digamos, Story Z.
 - 1.1 La story a ser estimada se lee a todo el equipo.
 - 1.2 Los estimadores discuten la story, haciendo preguntas al product owner (las que se necesiten).
 - 1.3 Cada estimador selecciona una carta y pone la carta boca abajo en la mesa.
 - 1.4 Cuando todos pusieron las cartas, las mismas se exponen al mismo tiempo.
 - 1.5 Si todos los estimadores selecciona el mismo valor, ese es el estimado. Sino, los estimadores discuten sus resultados, poniendo especial atención en los más altos y los más bajos. Después de la charla, GOTO to 1.3.
2. Se toma la próxima story, se discute con el product owner.
3. Cada estimador asigna a la story un valor por comparación contra la base story. "Cuan grande/pequeña, compleja, riesgosa es esta story comparada con Story Z?". GOTO to 1.3

